



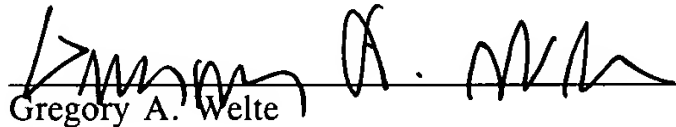
IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF APPEALS

Assignee's Docket No.: 9399.00 )  
Group Art Unit: 2176 )  
Serial No.: 09/704,066 )  
Examiner: Rachna Singh )  
Filing Date: November 1, 2000 )  
Title: Defining a Process by a )  
Plurality of Pages in a )  
Mark-up Language )

---

CERTIFICATE OF MAILING

I certify that this document is addressed to Mail Stop AF, Commissioner of Patents, PO Box 1450, Alexandria, VA 22313-1450, and will be deposited with the U.S. Postal Service, first class postage prepaid, on June 18, 2007.

  
Gregory A. Welte

CORRECTED APPEAL BRIEF  
A Summary of Argument Begins on Page 19

The Fee for this Brief has been paid.

1. REAL PARTY IN INTEREST

NCR Corporation.

2. RELATED APPEALS AND INTERFERENCES

None.

09/704,066  
Art Unit 2176  
9399.00

**3. STATUS OF CLAIMS**

Claims 1 - 27 are pending, rejected, and appealed.

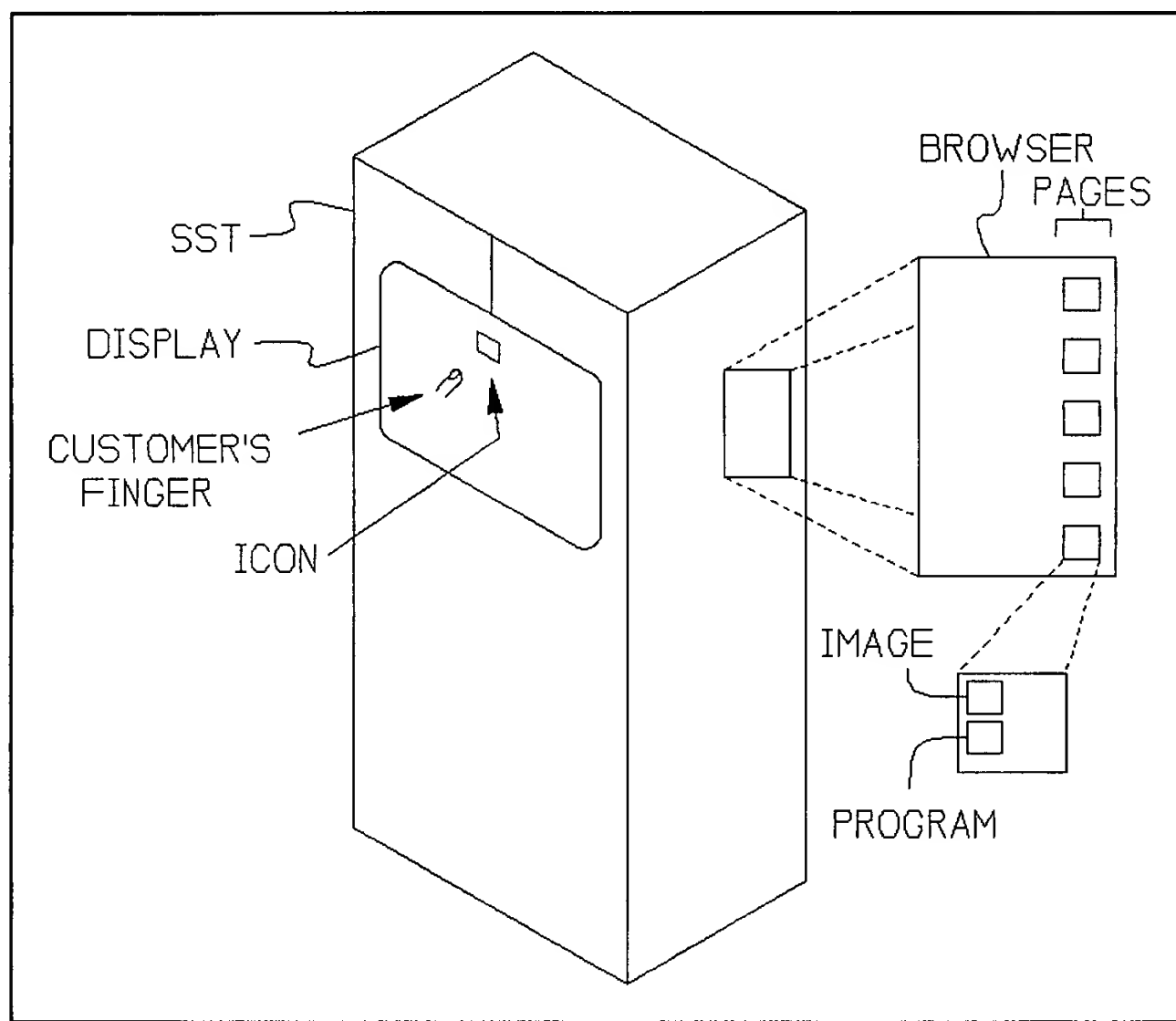
**4. STATUS OF AMENDMENTS**

No Amendments-After-Final have been submitted.

## 5. SUMMARY OF CLAIMED SUBJECT MATTER

### Basics of Invention

Sketch 1, below, illustrates a Self-Service Terminal, SST, Which may be an Automated Teller Machine, ATM.



SKETCH 1

The invention maintains a BROWSER within the SST. The BROWSER is a conventional web browser, which one uses to surf the Internet.

The BROWSER maintains access to PAGES, which are analogous to web pages. Each PAGE is a data block, and includes, for example,

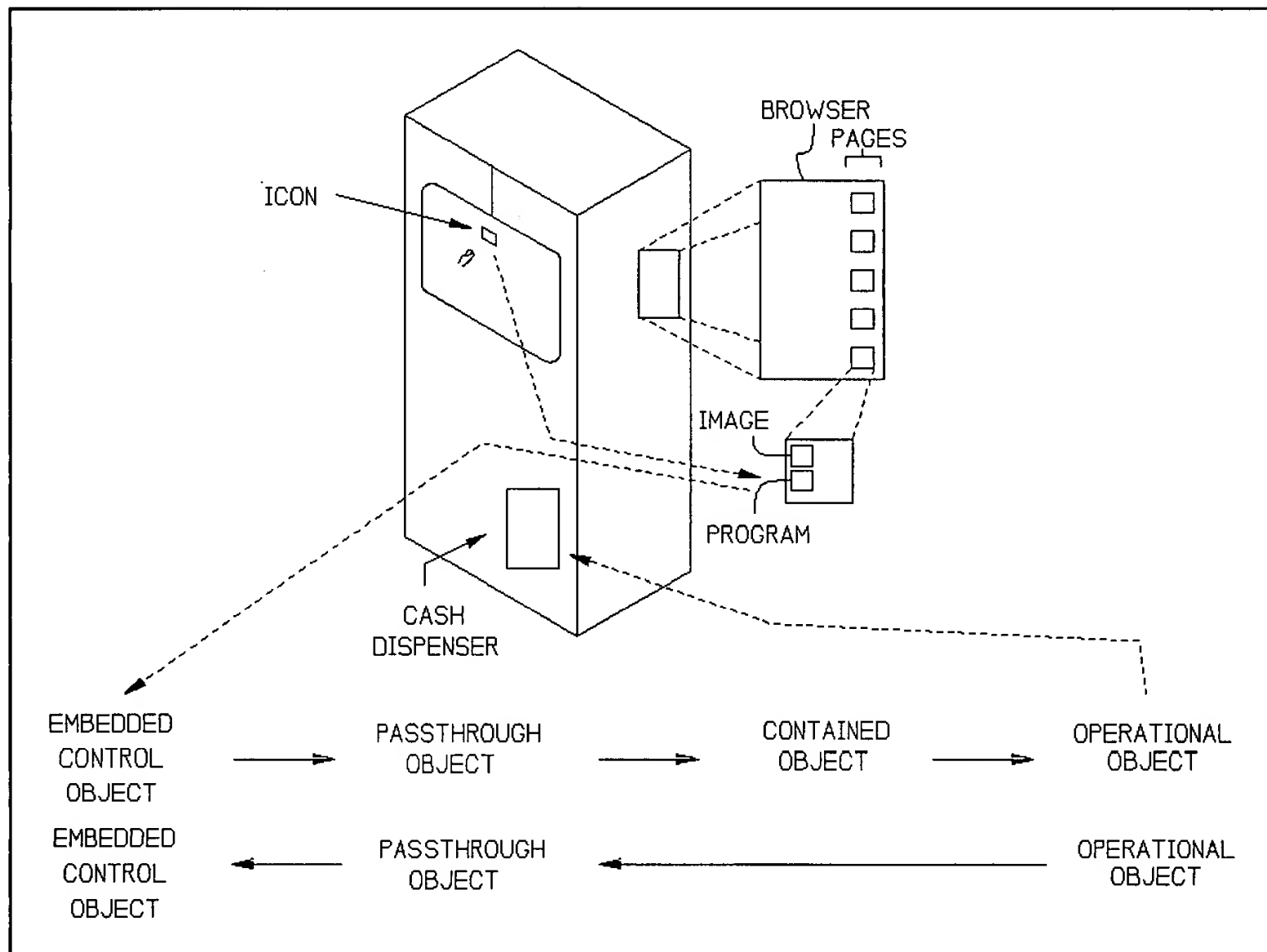
09/704,066  
Art Unit 2176  
9399.00

- 1) graphical data, labeled IMAGE, which is used to generate a visual display, including a displayed ICON which a customer can actuate, and
- 2) PROGRAM(s), also called "objects," which are called into action, when a user actuates the ICON.

To repeat: in SKETCH 1, the BROWSER generates an image on the DISPLAY. The image includes an ICON. A customer (not shown) can actuate the ICON by pressing the ICON with the CUSTOMER'S FINGER.

SKETCH 2, below, illustrates operations which the invention undertakes in response to actuation of the ICON by the CUSTOMER'S FINGER.

09/704,066  
 Art Unit 2176  
 9399.00



SKETCH 2

Assume that the ICON calls for a withdrawal of one hundred dollars. Actuation of the ICON causes the PROGRAM associated with the PAGE which is displayed to be launched, as indicated by the dashed arrow pointing to the PROGRAM. The ICON-PROGRAM combination, in the terminology of claim 1 (Specification, page 14, lines 1 - 21), is called an "embedded control object," because "embedded" in the page. A dashed arrow points to the "embedded

09/704,066  
Art Unit 2176  
9399.00

control object."

The sequence immediately below the SST ensues, as claim 1, line 14 et seq., indicates.

-- The "embedded control object" calls a  
"passthrough object."

-- The "passthrough object" calls a  
"contained object."

-- The "contained object" calls an  
"operational object."

-- The "operational object" actuates the CASH  
DISPENSER.

Then the "operational object" returns data, as indicated in the sequence at the very bottom of SKETCH 2. The data may indicate, for example, whether or not the customer took the cash which was dispensed. (See Specification, page 7, lines 11 - 17, and steps such as step 301 in Figure 3.)

The returning data is delivered to the "passthrough object" and then to the calling "embedded control object."

Notice that, under claim 1, the returning data is not delivered to the "contained object."

The procedure just described is undertaken because use of the BROWSER can create a specific problem. After actuating the ICON, the customer may "surf" to a different PAGE. If that occurs before the returning data reaches the "embedded control object" in the

09/704,066  
Art Unit 2176  
9399.00

last sequence of SKETCH 2, that data can be lost, thereby causing malfunctions.

The following discussion will explain this type of problem in greater detail.

### **Background**

The invention utilizes a "web browser" within an Automated Teller Machine, ATM. The web browser itself is a standard item, and is of the type used to surf the Internet. (Specification, page 5, lines 3 - 5; page 2, lines 16, 17.)

The web browser

- 1) acts as an interface with customers  
and
- 2) controls operation of mechanical devices  
within the ATM.

The web browser provides several advantages, including the following:

- 1) The effort to design the browser was undertaken by a third party. Thus, the ATM owner need not "re-invent the wheel," so to speak, by designing an equivalent system from scratch.
- 2) Customers are familiar with such browsers, through usage of the Internet, so they need

09/704,066  
Art Unit 2176  
9399.00

not re-learn a new interface.

3) The browsers have been extensively tested, bugs have been worked out, and they are thus very robust.

### **Problem**

A problem with the use of a web browser can be explained with reference to Figure A of the Specification. Blocks A1, A2, and A3 indicate three different web pages which can be displayed by an ATM, at different times.

While it is common to consider a web page as a visual image, in fact, the visual image is actually derived from a block of data, which contains additional elements. The additional elements include (2) computer programs and (3) other material. (Specification, page 1, line 26 et seq.) An example can illustrate such other elements.

The visual image which is displayed may include an icon labeled "\$ 100 withdrawal." A computer program is embedded, or associated with, the block of data from which the visual image was derived. When a customer actuates the icon, the computer program is launched, and causes a dispensing mechanism to deliver one hundred dollars to the customer.

In Figure A, block A4 can represent such an icon. When the customer selects block A4, the computer process indicated by arrow

09/704,066  
Art Unit 2176  
9399.00

A6 is initiated. This computer process A6 actuates the dispensing mechanism. After dispensing, certain data will return to the computer process, as indicated by arrow A7. The data may indicate that \$ 100 was dispensed, so that the web page A1 can then take accounting steps, or cause other programs to take the steps, in order to rectify the customer's balance.

Now the problem can be explained.

If a new web page A2 is displayed, before the data of arrow A7 returns, many browsers cannot handle that returning data when it arrives. A basic reason is that the computer program which processes the data indicated by arrow A7 is associated with web page A1. When web page A2 is displayed, page A1 disappears, and the computer program associated with it also disappears.

A more specific example can illustrate the problem further. Arrow A7 in Figure A can represent the transmission of a specific variable from the dispensing mechanism to the computer program under discussion. The variable may be

"MONEY\_DISPENSED = \$100."

The computer program in question will recognize that variable.

However, there is no computer program present within web page A2 which can (1) recognize this variable, nor (2) process the numerical amount within the variable. Thus, the variable will not

09/704,066  
Art Unit 2176  
9399.00

be processed by web page A4.

A rough analogy can further illustrate the nature of the problem. We know how to order merchandise over the Internet. After we select merchandise, a web page asks for a credit card number, and ordinarily we provide it. However, if we instead enter "Abe Lincoln" into the blanks for the credit card number, the web page cannot handle that.

Similarly, web page A2 in Figure A cannot handle the variable "MONEY\_DISPENSED = \$100" when it arrives. (Specification, page 2, line 10 - page 3, line 3.) Web page A2 contains no lines of code which understand this variable.

### **Invention**

During a transaction with a customer at an ATM, the web browser discussed above selects a sequence of web pages, and displays them to the customer. Different events occur in each web page.

For example, on one page, the customer may request that one hundred dollars be dispensed. In response to the request, computer code, associated with the web page, will be called up which activates the dispenser mechanism. The dispenser may return a dispense signal stating that the currency was successfully dispensed. Also, the dispenser may detect whether the currency was physically withdrawn by the customer, or left in place, and issue

09/704,066  
Art Unit 2176  
9399.00

a removal signal, or non-removal signal, accordingly. (See Specification, page 7, line 13 et seq. and Figure 3, block 309 et seq.)

However, the invention does not rely on the individual pages, by themselves, to handle the dispense and removal signals (for example). One reason is that, as explained above, if a page is not active when such signals arrive, the signals are not processed.

That is, as explained above, if the customer requested a withdrawal, and then moved to another page before the dispense and removal signals arrived, those signals would be lost. Other software which relies on those signals would then perhaps not function properly.

Instead, the invention creates an intermediate agent, called a "passthrough object," which handles the situation, and records the signals received. In general, the "passthrough object" stores the signals in question for the benefit of programs needing the signals.

For example, as explained in the Specification, page 9, lines 16 - 21, if an event, such as that indicated by arrow A7 in Figure A, returns, but the page responsible for the event is not longer active (ie, not "registered"), the returning event is stored until that page becomes active again. (See also Specification, page 10, lines 7 - 11.)

The Specification, page 11, line 12 - page 12, line 15

09/704,066  
Art Unit 2176  
9399.00

provides an example, in the context of a cash withdrawal.

### **Explanation of Claim Terminology**

The discussion just given will be explained in terms of claim 1, which is set forth below. This explanation applies to the other claims. Comments are added to claim 1, in parenthetical **Times Roman Bold Type**. Also, some comments clarify standard terminology which is used by computer scientists.

1. Data processing apparatus having processing means, memory means and display means, wherein

said processing means performs a process in response to program instructions read from said memory means via dynamically linked operational objects called by control objects, such that events are returned back to a calling control object;

("Processing means" can refer to the microprocessor in a computer. "Control objects" and "operational objects" can refer to visual "buttons," or icons, on a displayed page, which the user actuates, and which cause associated code to be executed. In a strict sense, the "objects" are the code. But, since the user does not actually see the code, conceptually, the "buttons" represent the action of the code, and are sometimes called "links.")

a plurality of pages are defined in a mark-up language that are selectively displayed and executed by a controlled browser;

(For example, the mark-up language (1) determines how the pages are displayed, as by controlling formatting, and (2) allows the "buttons," also called "links" to be embedded in the pages. But, again, actual "linking" is caused by code, but appears to be caused by the "buttons.")

said controlled browser is controlled by a controlling container object;

09/704,066  
Art Unit 2176  
9399.00

active control objects for calling said operational objects are contained within said container object;

**(For example, the buttons displayed on the pages, and the codes which the buttons call when actuated, are contained in the container object.)**

a single pass-through object is created;

**(Note that a "single" pass-through object is created.)**

at least one of said pages includes a page embedded control object configured to call said passthrough object;

**(For example, when a button is pressed, the passthrough object is called. The button, its associated code, or both are the "page embedded control object." )**

an initiating one of said page embedded objects calls said passthrough object and passes to said passthrough object output information detailing a desired call to a specified operational object;

**(For example, rather than calling an "operational object" directly when a button is pressed, the desired "operational object" is identified to the passthrough object. The passthrough object then calls the operational object.)**

said passthrough object interprets output information received from a page embedded object to generate a call to a contained object that in turn calls the desired operational object; and

**(Again, the passthrough object acts as an intermediary.)**

said passthrough object receives event data from a called operational object and returns input data to said initiating embedded object indicative of said returned event.

09/704,066  
Art Unit 2176  
9399.00

(Again, the passthrough object acts as intermediary.)

In addition, although claim 1 does not emphasize this, the passthrough object can store data, such as data generated by the called code, in a buffer. (See Specification, page 9, line 6 et seq. See claims 8 - 11.) Thus, if a page which initiated a call is not longer displayed, the returned data is not lost. That shows one useful feature of the invention.

#### **Mapping of Claim Elements to Specification and Figures**

Parenthetical phrases, in **bold typeface**, are inserted into the following independent claims, to identify matter in the Specification and Figures which supports the claim language adjacent said **bold, parenthetical typeface**.

1. Data processing apparatus (**SST 101, Figure 1; page 5, line 14**) having processing means, memory means and display means, wherein

said processing means (**central processing unit 201, Figure 2; page 6, lines 1, 2**) performs a process in response to program instructions (**page 6, lines 4 - 10**) read from said memory means (**system memory 202, Figure 2; page 6, lines 2 - 5**) via dynamically linked operational objects called by control objects

09/704,066  
Art Unit 2176  
9399.00

(soft keys 103, Figure 1; page 5, lines 17 - 20; page 7, lines 6 - 9; operational objects 411, Figure 4), such that events are returned back to a calling control object;

a plurality of pages are defined in a mark-up language that are selectively displayed and executed by a controlled browser (page 8, lines 12 - 20; blocks 301, 303 in Figure 3; browser control 409, Figure 4; page 10, line 7);

said controlled browser is controlled by a controlling container object (page 10, lines 7,8; controlling container object 410, Figure 4);

active control objects for calling said operational objects are contained within said container object (active control objects 411, Figure 4; page 10, lines 8, 9);

a single pass-through object is created (page 10, lines 9, 10; passthrough object 431, Figure 4);

at least one of said pages includes a page embedded control object configured to call said passthrough object (page 10, lines 10 - 14);

an initiating one of said page embedded objects calls said passthrough object and passes to said passthrough object output information detailing a desired call to a specified operational object (page 10, lines 12 - 14);

said passthrough object interprets output information received from a page embedded object to generate a call to a

09/704,066  
Art Unit 2176  
9399.00

contained object that in turn calls the desired operational object (page 10, lines 15 - 17; contained object 417, Figure 4; page embedded object 423, Figure 4; desired operational object 441, Figure 4); and

said passthrough object receives event data from a called operational object and returns input data to said initiating embedded object indicative of said returned event (page 10, lines 17 - 19; step 301 in Figure 3).

12. A method of defining a process in a computer system via a plurality of pages defined in a mark-up language that are executable by a browser (page 8, lines 12 - 20; page 10, lines 3 - 5), in which

said process implements operations via dynamically linked operational objects called by control objects such that events are returned back to a calling control object (soft keys 103, Figure 1; control objects 411 - 417, Figure 4; page 10, lines 12 - 14; page 10, lines 4 - 6);

a plurality of pages defined in a mark-up language are selectively displayed and executed by a controlled browser ("pages" 406, 407, 408, Figure 4; page 10, lines 7, 8);

said controlled browser is controlled by a controlling container object (page 10, line 8; page 8, lines 16 - 18);

active control objects for controlling said operational

09/704,066  
Art Unit 2176  
9399.00

objects are contained within said container object (page 10, lines 8, 9; active control objects 411 - 417, Figure 4);

a single passthrough object is created (page 10, lines 9, 10; passthrough object 431, Figure 4); and

pages include a page embedded control object configured to call said passthrough object, said method comprising the steps of (page 10, lines 10 - 12):

calling said passthrough object from an initiating embedded object and passing output information detailing a desired call to a specified operational object (page 10, lines 12 - 14);

interpreting said output information at said passthrough object to effect a call to said operational object (page 10, lines 15 - 17; operational object 441, Figure 4); and

returning input data to an initiating embedded object indicative of a returned event via said passthrough object (page 10, lines 17 - 19; initiating embedded object 423, Figure 4; step 301 in Figure 3).

21. A computer-readable medium having computer-readable instructions executable by a computer such that, when executing said instructions, said computer will perform the steps of:

establishing a library of dynamically linkable objects that may be called by control objects such that events are returned back to a calling control object (soft keys 103, Figure 1; control

09/704,066  
Art Unit 2176  
9399.00

objects 411 - 417, Figure 4; page 10, lines 12 - 14; page 10, lines 4 - 6);

establishing the availability of a plurality of pages defined in a mark-up language that may be selectively displayed and executed by a controlled browser, wherein said controlled browser is controlled by a controlling container object ("pages" 406, 407, 408, Figure 4; page 10, lines 7, 8) ;

containing active control objects for controlling operational objects within said container object (page 10, lines 8, 9; active control objects 411 - 417, Figure 4); and

facilitating the establishment of a single passthrough object (page 10, lines 9, 10; passthrough object 431, Figure 4), wherein pages defined in said mark-up language include a page embedded control object configured to call said passthrough object (page 10, lines 10 - 12) such that, during a session with a user, said program computer will further perform the steps of:

calling said passthrough object from an initiating embedded object and passing output information detailing a desired call to a specified operational object (page 10, lines 12 - 14);

interpreting said output information at said passthrough object to effect a call to said operational object (page 10, lines 15 - 17; operational object 441, Figure 4); and

returning input information to an initiating embedded object indicative of a returned event via said passthrough object

09/704,066  
Art Unit 2176  
9399.00

(page 10, lines 17 - 19; initiating embedded object 423, Figure 4; step 301 in Figure 3).

#### 6. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The rejections of claims 1 - 27 as obvious under 35 USC § 103, based on Shima '507.

#### 7. ARGUMENT

##### SUMMARY OF ARGUMENT

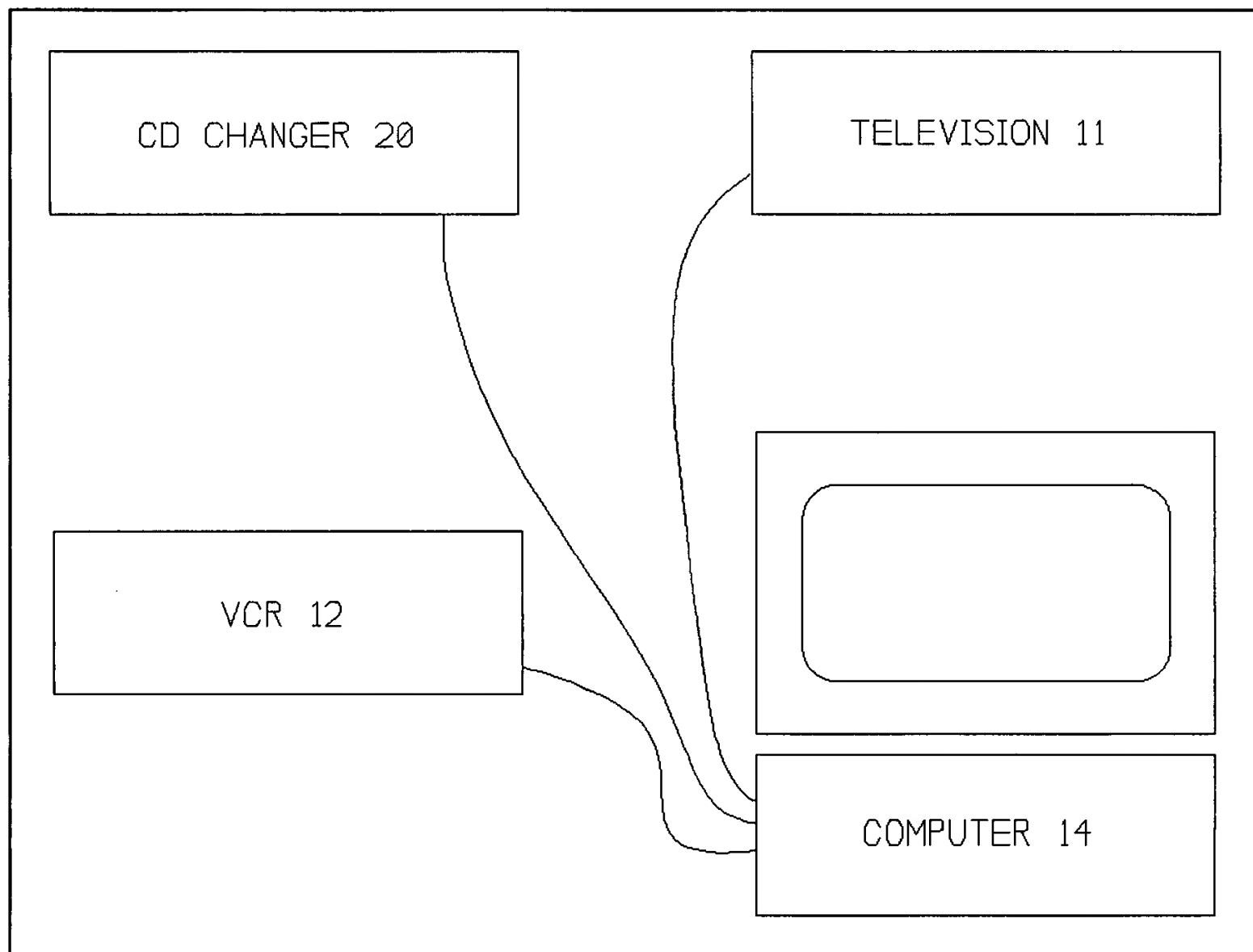
##### Shima Reference

Shima states that many people own different media devices, such as VCRs, video cameras, etc. (Column 1, lines 26 - 34.) Each device has dials and switches, or a "control panel," by which a user operates the device.

Shima proposes that a pictorial representation of the control panel of each device be stored within a computer. Thus, numerous different control panels are stored, one for each device. When the user wants to operate a particular device, the representation of the control panel for that device is retrieved and displayed. (Figure 9, blocks 712 and 714.)

SKETCH 3, below, is a rendition of Shima's Figure 1, and illustrates three devices connected to a computer, which controls the devices.

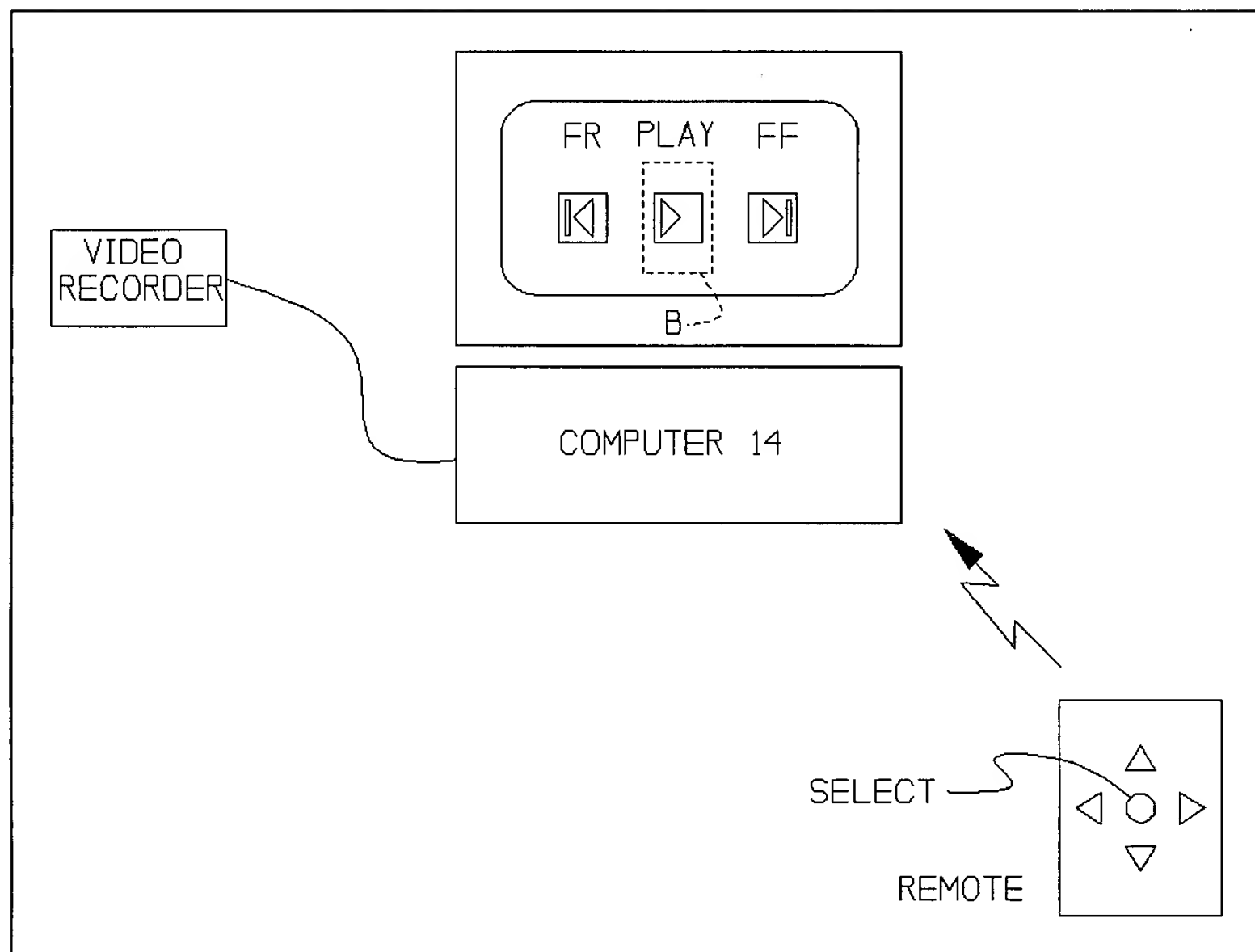
09/704,066  
Art Unit 2176  
9399.00



SKETCH 3

09/704,066  
Art Unit 2176  
9399.00

SKETCH 4, below, illustrates Shima's approach, and is based on his Figure 13A, and his column 20, line 45, et seq.



SKETCH 4

09/704,066  
Art Unit 2176  
9399.00

In SKETCH 4, an image of part of the control panel of a video recorder is displayed on a screen. Buttons FR, PLAY, and FF are shown, which refer to "Fast Rewind," "Play," and "Fast Forward," respectively.

Shima uses a REMOTE. Arrow-buttons on the REMOTE allow the user to highlight, as indicated by dashed box B, a button on the display. Button PLAY is shown highlighted.

After highlighting a button on the display (the PLAY button in this example) the user presses the SELECT button on the REMOTE, to actuate the selected (PLAY) button on the display. The video recorder then plays.

#### Comment on Shima

Appellant repeats that Shima generates a representation of a control panel for each of

- 1) a VCR,
  - 2) a video camera,
  - 3) a CD changer,
- etc. (Column 6, line 66 et seq.)

These representations will be called "virtual control panels."

Shima displays a control panel on a television, and a user operates a device, through the displayed control panel, using a REMOTE. (Column 10, lines 3 - 16.)

09/704,066  
Art Unit 2176  
9399.00

### Flaws in Obviousness Rejections

This discussion will consider claim 1, but the principles discussed apply to all claims.

#### FLAW 1

The Final Action is double-counting a **single** element in Shima, in order to show **two** claim recitations. That is not allowed. MPEP § 2143.03 states:

To establish prima facie obviousness . . . **all the claim limitations** must be taught or suggested by the prior art.

The Final Office Action (page 4, first bullet paragraph, beginning with "--An intelligent controller . . .") asserts that Shima's icons (or icon-program combinations), such as FR, PLAY, and FF in SKETCH 4, above, show the "control object" of claim 1.

The Final Action (page 5, first bullet paragraph, beginning with "-A panel subunit . . .") asserts that **these same icons** show the claimed "passthrough object," because commands "pass through" those icons, as it were.

The Final Action is double-counting a **single** element in the reference, to show **two** claim recitations. That **single** element is Shima's virtual control panel of Sketch 4, above.

Double-counting is not allowed.

09/704,066  
Art Unit 2176  
9399.00

## FLAW 2

Claim 1, line 14 et seq., recites a sequence of call-passing. SKETCH 2, above, illustrates the sequence.

Claim 1, line 17 et seq., recites a sequence of data-passing, with the data

- 1) originating in the object (or program)  
which executed (or performed) the call and
- 2) being delivered to the object (the  
"embedded control object") which initiated the  
call.

SKETCH 2, above, illustrates the data-passing sequence.

These two sequences have not been shown in Shima.

The Final Action, page 5, second bullet-paragraph (beginning with "-The pass-through command . . .") attempts to show these two sequences. However, the Final Action merely asserts that certain types of operation **may be** present in Shima, but does not actually show those operations.

Further, the actual claim language has not been shown in Shima. Instead, the Final Action, in the paragraph in question, purports to characterize Shima, and then implies, by stating "compare with," that the characterization is similar to the claim language.

That is insufficient. In effect, that re-writes Appellant's

09/704,066  
Art Unit 2176  
9399.00

claim, and then applies the claim, as re-written, to Shima.

Appellant's claim language itself must be shown in Shima. That has not been done. The two sequences described above have not been shown in Shima.

### FLAW 3

This is a specific example of FLAW 2. As SKETCH 2, above, indicates, data is returned from the "operational object" to the "embedded control object." As stated above, the data may indicate whether dispensed cash has been removed by a customer.

No such "data" has been shown in Shima.

Further, such data appears to make no sense in the context of Shima. Assume, for example, that the user presses SELECT in Sketch 4, above, to cause Shima's video recorder to play.

-- What data would be returned to Shima's  
display in SKETCH 4 ?

-- Where does Shima discuss such data ?

### FLAW 5

The Final Action asserts that all recitations of claim 1 are found in Shima, with the exceptions that

- 1) Shima's "pages" are not in a "mark-up  
language" as claimed  
and

09/704,066  
Art Unit 2176  
9399.00

2) Shima does not use a browser.

The Final Action then asserts that these missing elements are "obvious."

However, "obviousness" is not a vehicle for supplying missing claim elements. Only **claims** can be obvious. Missing elements are not "obvious."

Missing elements may be Officially Noticed, to eliminate a need for documenting them. But missing elements are not "obvious."

Further, the Final Action has not shown how, or why, Shima would want to add (1) "pages" in a "mark-up language" and (2) a browser.

Further still, the Final Action has not shown how a browser would be installed in Shima, to accomplish what claim 1 recites. No expectation of success, as required by the MPEP and discussed later, has been shown.

#### FLAW 6

The PTO asserts that certain claim elements (the mark-up language and the browser) are not shown in Shima. But no valid teaching has been given for combining the missing elements with Shima.

One rationale given (Final Action, page 6) employs circular logic. That rationale is that it is obvious to add these two elements to Shima, because a browser reads pages written in a mark-

09/704,066  
Art Unit 2176  
9399.00

up language.

That is a non-sequitur: the conclusion does not follow from the premise. The premise is that browsers read pages. The conclusion is that browsers and pages should be added to Shima, because of the premise. That is a non-sequitur.

The fact that a browser reads pages does not lead to the conclusion that a browser of the type recited in claim 1 should be added to Shima.

Another rationale given is that Shima teaches the use of a browser, so that Shima thereby teaches that "pass-through functionality" should be implemented using that browser. However, several problems exist in this rationale.

#### FLAW 6 - PROBLEM 1

Even if the rationale be valid, it does not show claim 1. Claim 1 does not recite mere "pass-through functionality," but the use of a "passthrough object" as claimed in claim 1, and such usage is represented in SKETCH 2, above. Thus, even if the rationale supports the use of "pass-through functionality," that does not show claim 1.

From another perspective, the rejection is incomplete. The rejection fails to show that the "pass-through functionality" supposedly suggested by Shima corresponds to the recitations of claim 1.

09/704,066  
Art Unit 2176  
9399.00

#### FLAW 6 - PROBLEM 2

The Final Action ignores at least two crucial facts.

One fact is that Shima, column 18, lines 33 - 37, states that the "target device" can be a computer which can surf the Internet. That is, the VIDEO RECORDER of SKETCH 4, above, becomes the web-surfing computer.

The Final Action, pages 6 and 11, interprets this as showing a "browser" in Shima. However, even if so, that "browser" does not control any devices. That "browser" is **within a CONTROLLED DEVICE** in Shima.

Restated, such a "browser" does not generate "pages" as in claim 1, and described in greater detail later.

Further, such a "browser" does not create an "embedded control object" as claimed, which is involved in the two sequences of SKETCH 2, above, as claimed.

Such a "browser" is not involved in controlling the SST, as recited in claim 1.

The second crucial fact is that, as just explained, the "browser" supposedly found in Shima is **already there**, by the PTO's reasoning. That is, according to the PTO, Shima already shows a browser.

But that browser does not perform as claim 1 recites.

Therefore, the PTO is

09/704,066  
Art Unit 2176  
9399.00

- 1) modifying Shima's (pre-existing) browser,  
in order to make it correspond to claim 1  
but
- 2) the modifications have not been shown in  
the prior art.

This approach of the PTO is not sanctioned by the MPEP. MPEP  
§ 2143.01 states:

THE PROPOSED MODIFICATION CANNOT RENDER THE  
PRIOR ART UNSATISFACTORY FOR ITS INTENDED  
PURPOSE.

THE PROPOSED MODIFICATION CANNOT CHANGE THE  
PRINCIPLE OF OPERATION OF A REFERENCE.

MPEP § 2143.03 states:

To establish prima facie obviousness . . . **all  
the claim limitations** must be taught or  
suggested by the prior art.

#### FLAW 7

Claim 1 recites a **single** "passthrough object." The Final  
Action asserts that the virtual control panel of Shima corresponds  
to the "passthrough object."

But Shima shows **multiple** virtual control panels, one for every  
device controlled.

Thus, Shima is **directly contrary** to claim 1 on this matter.

09/704,066  
Art Unit 2176  
9399.00

**Comment**

Not all points made in this Summary of Argument are elaborated below. Some are considered self-explanatory.

This Summary applies to the other independent claims, namely, claims 12 and 21.

**END SUMMARY OF ARGUMENT**

09/704,066  
Art Unit 2176  
9399.00

## ARGUMENT

### GROUP 1: CLAIMS 1 - 11

#### Claim 1

Claim 1 will be considered.

#### Point 1

Claim 1 recites:

a plurality of pages are defined in a  
mark-up language that are selectively  
displayed and executed by a controlled  
browser.

To repeat, this claim recitation states, inter alia, that

- "pages" are present,
- the "pages" are defined in a "mark-up  
language," and
- the "pages" are executed by a "browser."

The Final Action, page 4, third paragraph, asserts that these  
claim recitations are found in Shima because Shima shows

A user interface implemented within the  
intelligent controller that is coupled within  
a networked system and has basic input and  
display capabilities.

However, these elements, even if present in Shima, do not  
show the claim recitations in question, for several reasons.

09/704,066  
Art Unit 2176  
9399.00

REASON 1

One reason is that no "browser" as claimed has been shown in Shima.

REASON 2

A second reason is that the mere presence in Shima of "a user interface" having "basic input and display capabilities" does not show the "pages" as recited.

REASON 3

A third reason is that no "mark-up language" as claimed has been shown in Shima.

Appellant points out that "mark-up language" is a term-of-art. The Specification, page 1, line 10 et seq., sets forth one feature of a "mark-up language," namely, the ability to provide "sophisticated graphical displays."

One way the "mark-up language" does this is to allow insertion of special codes within a text document. The codes themselves are not displayed, but determine, for example, how nearby text is displayed. Or, as another example, the code defines a graphical image, which is displayed.

CONCLUSION AS TO POINT 1

MPEP § 2143.03 states:

09/704,066  
Art Unit 2176  
9399.00

To establish prima facie obviousness . . . **all the claim limitations** must be taught or suggested by the prior art.

The claim recitation in question has not been shown in Shima.

#### Point 2

The Final Action is self-contradictory.

-- On page 4, second paragraph (beginning with "-- An intelligent controller . . .), the Final Action asserts that the claim recitation of Point 1, above, is shown in Shima. As explained above, that recitation recites "pages" in a "mark-up language" and a "browser."

-- On page 5, last line et seq., the Final Action states that Shima **DOES NOT** show pages in a mark-up language, nor a browser.

Thus, the Final Action is self-contradictory.

Self-contradictory arguments are invalid, as a matter of logic. They do not lead to a valid conclusion.

Therefore, the conclusion of obviousness of the Final Action is invalid.

09/704,066  
Art Unit 2176  
9399.00

Point 3

Claim 1 recites a "passthrough" object, which performs specific functions, such as those set forth in the last four sub-paragraphs of claim 1.

The Final Office Action, page 5, bottom, asserts that Shima shows "pass-through functionality," because Shima's "sub panels" (ie, the virtual control panels) accept commands, and pass the commands to a device.

However, claim 1 does not recite mere "pass-through functionality." It recites a "passthrough object," and recites specific actions which that "object" takes.

For example, claim 1, last four sub-paragraphs, states that **bi-directional** "passing" occurs. That is, claim 1 states that an "embedded object" can initiate a "call." (A computer program, or sub-routine, can be termed an "object." A "call" is a command to execute an object.)

After the "call" is initiated by the "embedded object," the "passthrough object"

-- handles the call when en route **TO** an  
"operational object" which executes the call,  
and

-- handles "event data" **FROM** the  
"operational object" which is returning to the  
"initiating embedded object."

09/704,066  
Art Unit 2176  
9399.00

That **bi-directional passing** has not been shown in Shima.

Therefore, a "passthrough object" as claimed, and its claimed functions, have not been shown in Shima.

#### Point 4

Claim 1 recites a "single" "passthrough object." The Specification, page 12, line 22 et seq., explains that the **single** "passthrough object" keeps track of which web pages are active.

Under the PTO's interpretation of Shima, **multiple** entities having "pass-through functionality" are present. That is, Shima expressly states that multiple representations of control panels are generated, one for each of several different devices.

Thus, Shima is **directly contrary** to this recitation of claim 1.

#### Point 5

The Final Action, page 5, last line et seq., admits that Shima fails to show the claimed "browser," "pages" and "pages in mark-up language." The PTO then argues that it is obvious to add these elements to Shima.

However, those elements have not been shown in the prior art. MPEP § 2143.03 states:

09/704,066  
Art Unit 2176  
9399.00

To establish prima facie obviousness . . . **all the claim limitations** must be taught or suggested by the prior art.

Merely asserting that such elements are obvious is insufficient. The elements must be shown in the prior art.

From another perspective, missing elements of a claim (as opposed to the entire claim) cannot be "obvious." Only "inventions" can be "obvious."

"Obviousness" is not a device to supply elements missing from the prior art, in order to support a rejection. This perspective is explained in greater detail in Point 6, below.

#### Point 6

The PTO mis-applies the doctrine of obviousness.

In essence, the PTO asserts that several elements of claim 1, which have **not been shown** in the prior art, are "obvious." However, under section 103, the question is whether an "invention" is "obvious," in view of the prior art. The question **is not** whether individual elements of an invention are "obvious."

Thus, (1) the claim elements must be shown in the prior art and (2) a determination must be made as to whether the elements, as claimed, are "obvious."

The doctrine of obviousness is not a substitute for showing all claim elements within the prior art, as required by the MPEP

09/704,066  
Art Unit 2176  
9399.00

section cited in Point 1, above.

#### Point 7

The rejection merely asserts that it is "obvious to extend Shima's system to include" the missing elements. (Final Office Action, page 6, first full paragraph.)

However, merely including those elements in Shima's system does not necessarily show claim 1. For example, where are the added "pages" specifically positioned within Shima's system, so that they function as claim 1 recites ?

Stated another way, the PTO has merely set forth the naked conclusion that, if Shima is "extended," then claim 1 is found. That is insufficient under section 103.

In fact, this conclusory rejection fails to comply with MPEP § 706.02(j), which requires an explanation in support of the rejection, and which states:

#### Contents of a 35 U.S.C. 103 Rejection

. . . the examiner should set forth in the Office action:

(A) the relevant teachings of the prior art relied upon, preferably with reference to the relevant column or page number(s) and line number(s) where appropriate,

(B) the difference or differences in the claim over the applied reference(s),

(C) the proposed modification of the applied

09/704,066  
Art Unit 2176  
9399.00

reference(s) necessary to arrive at the claimed subject matter, and

(D) an explanation why one of ordinary skill in the art at the time the invention was made would have been motivated to make the proposed modification.

The rejection fails to set forth the "explanation" required by paragraph (D), above. Instead, the rejection merely asserts that it is "obvious" to make the "proposed modification" of paragraph (C), above.

That is insufficient, under the express terms of this MPEP section.

#### Point 8

Claim 1 recites:

said controlled browser is controlled by  
a controlling container object.

Even if Shima is "extended" to include "pages" and a "browser," it has not been shown how, or why, that "extension" would include this recitation.

Again, all claim recitations must be shown, as required by the MPEP, as explained above.

#### Point 9

Claim 1 recites:

09/704,066  
Art Unit 2176  
9399.00

at least one of said pages includes a  
page embedded control object configured to  
call said passthrough object.

The PTO has not shown that the "pages" which it adds to Shima  
have this claimed property.

That is, the PTO has not shown that the "pages" added to Shima

1) contain an "embedded control object,"

nor

2) that said "object" calls the "passthrough  
object."

Thus, again, all claim elements have not been shown in the  
prior art, contrary to the MPEP section cited above.

#### Point 10

This is a type of continuation of Point 9, above. The PTO has  
failed to show an expectation of success, as required by the MPEP.

The claim recitation of Point 9 is here repeated:

at least one of said pages includes a  
page embedded control object configured to  
call said passthrough object.

The Final Action asserts that Shima's display contains a  
visual icon which calls up a virtual control panel. The Final  
Action then asserts the following correspondence:

09/704,066  
Art Unit 2176  
9399.00

TABLE A

Shima	Claim 1
Visual icon	"Embedded control object"
Virtual control panel	"Passthrough object"

But that supposed correspondence exists **before** the missing "pages" are combined with Shima. That is, according to the PTO, TABLE A is found in Shima, prior to addition of the missing "pages."

The PTO has not explained how the now-added "pages" will contain the visual icon of Shima.

Thus, no expectation of success has been shown, indicating that the combination of references actually works.

MPEP § 706.02(j) states:

Contents of a 35 U.S.C. 103 Rejection

. . .

To establish a prima facie case of obviousness, three basic criteria must be met.

. . .

Second, there must be a reasonable expectation of success.

. . .

The . . . reasonable expectation of success must both be found in the prior art and not based on applicant's disclosure.

09/704,066  
Art Unit 2176  
9399.00

Point 11

MPEP § 2143.02 states:

The prior art can be modified or combined to reject claims as prima facie obvious as long as there is a **reasonable expectation of success**.

MPEP § 706.02(j) states:

. . . .

To establish a prima facie case of obviousness, three basic criteria must be met.

. . . .  
Second, there must be a **reasonable expectation of success**.

. . . .  
The . . . reasonable expectation of success must . . . be found in the prior art and not based on applicant's disclosure.

As explained above, the PTO has **failed to show** several elements of claim 1 in the prior art. Instead, the PTO has summarily concluded that it is obvious to extend Shima to include those missing elements.

Thus, the PTO has not actually shown all elements of claim 1 in the prior art. Consequently, no expectation of success has been shown, since the entirety of claim 1 has not been shown in the prior art.

That is, since the entirety of claim 1 has not been actually shown in the prior art, no expectation of success has been given

09/704,066  
Art Unit 2176  
9399.00

that the entire claim 1 will be attained in the prior art, even if combined.

Point 12

No teaching has been given in favor of the modification of Shima.

Sub-Point 12A

The rationale given by the PTO asserts that it is obvious to add a "browser" to Shima. But, as just explained, an expectation of success is required.

When you add that "browser," what does the browser browse through ? That is, in Shima, what does the browser look at ?

Until that question is answered, no expectation of success has been shown.

Sub-Point 12B

The rationale of the PTO asserts that it is obvious to also add "pages" to Shima. Where is the expectation of success ?

That is, what is the content of the pages ? Until that question is answered, no expectation of success has been shown.

Sub-Point 12C

The PTO, in essence, asserts that it is obvious to add a

09/704,066  
Art Unit 2176  
9399.00

"browser" and "pages" to Shima, because Shima shows a "networked system."

However, if Shima shows a "network," that is not the type of network that a browser uses. For example, Shima's Figure 1 shows

a computer 14,  
a VCR 12,  
a video camera 10,  
a compact disc (CD) changer 20,  
a set-top box (STB) 13, and  
a television 11.

He shows cables 15, 16, 18, and 19 connecting some of those elements together.

Appellant asks, how would a "browser" be added to those elements ? The only element on which a "browser" can run is the computer 14.

Assume arguendo that a "browser" is installed on that computer 14.

- How does it "browse" the television 11 ?
- How does it "browse" the STB 13 ?
- How does it "browse" the CD changer 20 ?
- How does it "browse" the VCR 12 ?

The answer to all questions is "It cannot."

Thus, no expectation of success has been shown.

09/704,066  
Art Unit 2176  
9399.00

Point 13

Claim 1 recites:

a plurality of pages are defined in a  
mark-up language that are selectively  
displayed and executed by a controlled  
**browser.**

Appellant points out that the PTO relies on Shima's implied  
browser of column 18, lines 34 - 37.

But claim 1 states that the "browser" generates "control  
objects," which perform specified functions. That has not been  
shown in Shima.

Point 14

Claim 1 recites:

at least one of said pages includes a  
page embedded control object configured to  
call said passthrough object.

This cannot be found in Shima, for at least the following reasons.

REASON 1

As explained elsewhere, claim 1 states that the "pages" are  
"defined in a mark-up language." The Final Action, page 4, third  
paragraph (beginning with "-A user interface . . ."), asserts that  
Shima's Figure 2 shows such "pages." However, no "mark-up

09/704,066  
Art Unit 2176  
9399.00

language" is shown.

Further, in a contradicting statement, the Final Action, page 5, last line, states that "pages" in a "mark-up language" are not present in Shima.

Thus, the claimed "pages" (in a "mark-up language") are not present in Shima.

## REASON 2

Under the PTO's interpretation of Shima,

- one of the icons in SKETCH 4, above, corresponds to the claimed "control object," and
- the display of SKETCH 4 shows the claimed "passthrough object."

However, this interpretation is impossible.

Under this interpretation, the claim phrase in question requires Shima's virtual control panel to call itself. That is, one of the buttons in SKETCH 4 will call the "passthrough object," which is supposedly already displayed.

Appellant submits that this interpretation is impossible.

Further, the PTO has not shown that a displayed button in Shima actually calls up anything corresponding to the claimed "passthrough object."

09/704,066  
Art Unit 2176  
9399.00

### Interim Conclusion

For the preceding reasons, Appellant submits that claim 1 is not obvious, based on Shima.

The preceding reasons apply to all other claims, some of which are discussed individually below.

### Response I to Final Action

Appellant, in his previous Amendment, page 13, under the heading "Sub-Point 6C" posed the question of how a browser would be added to Shima. Appellant further asserted that, since the question was not answered, no expectation of success in attaining claim 1 has been shown by the PTO, as required by the MPEP.

The Final Office Action, page 11, asserts that a "browser" could be added to Shima, because Shima, column 18, lines 25 - 37, asserts that his television can surf the Internet. (Id. at lines 34 - 37.)

However, that still does not show claim 1, for several reasons.

### REASON 1

One is that, at best, Shima merely asserts that an ordinary prior-art browser is added to his television. But claim 1 does not recite an ordinary browser. The browser in claim 1 contains the "passthrough object" and its special functions.

09/704,066  
Art Unit 2176  
9399.00

Thus, no expectation of success in attaining claim 1 is found in Shima.

#### REASON 2

Shima only states that his television, instead of a computer, is used to browse the Internet. (Apparently, his television is used as a computer monitor, and the computer is built into the television.) This television is then controlled by Shima's "control panel" in the manner described above.

That simply does not show claim 1. The missing elements, and problems, described above have not been shown, or resolved, by the PTO's reliance on Shima's special television.

#### REASON 3

As explained above, the Shima-passage in question states that the VIDEO RECORDER in SKETCH 4, above, becomes the device which surfs the Internet.

But that device is a **controlled** (not controlling) device. That device lacks the functions recited in claim 1.

Thus, claim 1 is not attained by Shima's web-surfer.

#### Response II to Final Action

The Final Action, page 11, only asserts that it is obvious to add

09/704,066  
Art Unit 2176  
9399.00

- a browser,
- pages in a mark-up language, and
- "pass-through functionality"

to Shima. However, even if that be done, claim 1 is still not attained.

The detailed operation and features, as outlined above, of claim 1 have not been shown in Shima, even as modified by the PTO's additions.

Further, the PTO asserts that "pass-through functionality" has been added to Shima. However, no "pass-through functionality" has been shown in the prior art. Nor has it been shown that such "functionality," even if present, corresponds to any recitation in claim 1.

#### Response III to Final Action

Claim 1, speaking generally, states that an embedded object specifies a call destined for an operational object. That call is given to the passthrough object, which calls a contained object, which then delivers the call to the operational object, which executes the call.

Then, the passthrough object receives output from the operational object, and relays the output to the original embedded object, which started the entire process.

Thus, with obvious abbreviations, the sequence of control runs

09/704,066  
Art Unit 2176  
9399.00

through objects in the following sequence:

Embedded -->> Passthrough -->> Contained -->> Operational.

Then, data returns in the following sequence:

Embedded <<-- Passthrough <<-- Operational.

Notice that the two sequences are not mirror-images. The "contained object" does not receive output from the operational object, and give it to the passthrough object. Instead, the passthrough object receives that output.

This type of operation is nowhere seen, nor even suggested, in Shima.

To support the preceding more specifically, Appellant points out that claim 1 states, inter alia, the following:

- an "embedded object" gives "output information" to a "passthrough object,"
- the "output information" "details" a desired call to an "operational object,"
- the "passthrough object" generates a call to a "contained object,"
- the "contained object" generates the call to the "operational object,"

09/704,066  
Art Unit 2176  
9399.00

-- the "passthrough object" receives output  
from the "operational object," and  
-- the "passthrough object" passes the output  
to the (originally calling) "embedded object."

Again, neither that actual operation, nor any reason for  
undertaking such operation, is found

#### Conclusion as to Claim 1

Appellant submits that the preceding discussion clearly shows  
that claim 1 is not shown, nor suggested, in Shima.

#### **Claims 2 and 3**

Claim 2 refers to a "self-service terminal," SST.  
Appellant points out that (1) "SST" is a term-of-art, (2) SSTs are  
contained in public places, and are used by the general public.  
In contrast, Shima refers to a home theater system. (Column 1.)  
He does not show a public place.

No SST has been shown in Shima.

Claim 3 depends from claim 2, and recites "wherein said self  
service terminal dispenses money and facilitates financial  
transactions." That has not been shown in Shima.

The PTO, bottom of page 4, asserts that Shima "**can** also  
include an ATM." That is insufficient as a rejection. MPEP §  
2143.01 states:

09/704,066  
Art Unit 2176  
9399.00

FACT THAT REFERENCES **CAN BE** COMBINED OR  
MODIFIED IS NOT SUFFICIENT TO ESTABLISH PRIMA  
FACIE OBVIOUSNESS

The mere fact that references can be combined  
or modified does not render the resultant  
combination obvious unless the prior art also  
suggests the desirability of the combination.

. . .

FACT THAT THE CLAIMED INVENTION IS WITHIN THE  
CAPABILITIES OF ONE OF ORDINARY SKILL IN THE  
ART IS NOT SUFFICIENT BY ITSELF TO ESTABLISH  
PRIMA FACIE OBVIOUSNESS

A statement that modifications of the prior  
art to meet the claimed invention would have  
been "well within the ordinary skill of the  
art at the time the claimed invention was  
made" because the references relied upon teach  
that all aspects of the claimed invention were  
individually known in the art is not  
sufficient to establish a prima facie case of  
obviousness without some objective reason to  
combine the teachings of the references.

Thus, the fact that Shima **could** be modified to include an ATM  
is not allowed as a basis for rejection by these MPEP sections.

Further, why would somebody put an ATM in Shima's environment,  
which is a living room of a home ?

In addition, the ATM used to reject claim 3 has not been shown  
in the prior art, contrary to MPEP § 2143.03.

#### **Claim 5**

Claim 5 recites:

09/704,066  
Art Unit 2176  
9399.00

5. Apparatus according to claim 1, wherein said passthrough object is a separate process executed under the control of an operating system.

The PTO relies on the "panel subunits" of Shima (ie, Shima's virtual control panels) to show the claimed "passthrough object." However, as explained above, those "subunits" involve multiple elements, both software and hardware. For example, the left side of Shima's Figure 13A shows a "panel subunit," containing hardware and included software.

That hardware/software does not correspond to the "passthrough object" of claim 5.

#### **Claim 6**

The rejection of claim 6 asserts that the mere presence of one single supposed "passthrough object" in Shima shows claim 6. That is not so.

Claim 6 recites an examination step, to determine whether such an object already exists. Shima does not perform that step.

#### **Claim 8**

Claim 8 states "wherein return events are returned to an initiating page via said passthrough object." The "return events" are, for example, generated by the "desired call to a specified

09/704,066  
Art Unit 2176  
9399.00

operational object" of claim 1.

Column 22, lines 19 - 24, of Shima could conceivably show part of claim 8. That passage states that the VCR generates a channel number which is displayed on the TV. That may be a "return event."

However, claim 8 states that the "events" are returned to "an initiating page via said passthrough object." Appellant points out that no "initiating page" has been identified in Shima, nor the passage of the "return event" through the "passthrough object" en route to that "initiating page."

#### **Claim 9**

The "register" of claim 9 has not been identified in Shima.

#### **Claims 10 and 11**

Dependent claim 10 states:

. . . wherein said passthrough object includes a buffer for buffering events to be returned to pages that are no long established within the process.

That is, claim 10 states that, when some pages are no longer established, the buffer holds events for those pages.

Those pages, and the buffering, have not been identified in Shima.

This applies to claim 11.

09/704,066  
Art Unit 2176  
9399.00

Dependent claim 11 states: ". . . buffered events are returned to non-established pages after said pages have been re-established."

Establishment, and non-establishment, of pages has not been shown in Shima, nor the claimed return of events, after re-establishment.

09/704,066  
Art Unit 2176  
9399.00

**GROUP 2: Claims 12 - 20**

**Claim 12**

As stated above, claims 12 - 20 are method claims, which are separately patentable from the remaining claims, which are apparatus claims.

Point 1

As to obviousness, the discussion of claim 1 applies to claim 12. That discussion is hereby incorporated by reference into this section.

Point 2

In addition, claim 12, line 13 et seq., state the following:

- "output information" originates with an "embedded object;"
- the "output information" is interpreted at a "passthrough object;"
- the interpretation effects a "call" to an "operational object;" and
- "input data" returns to the "embedded object" through the "passthrough object."

Also, line 10 states that a "single" passthrough object is created.

The Final Action relies on the rejection of claim 1 to reject

09/704,066  
Art Unit 2176  
9399.00

claim 12. The rejection of claim 1 asserted that Shima shows the claimed "passthrough object" in his virtual control panels (ie, his "subunits"). However, Appellant pointed out that Shima discusses **multiple** virtual control panels, one for each device controlled. That is contrary to the **single** "passthrough object" recited in claim 12.

#### Point 3

Claim 1 states that the "passthrough object" performs interpreting, to generate a "call" to a "contained object."

However, claim 12 states something different, namely, that the "passthrough object" performs interpreting, to generate a "call" to "effect a call to said operational object." In claim 12, the "call" is to a different "object."

Therefore, reliance on the rejection of claim 1 fails to show this different claim recitation in claim 12.

#### Point 4

Claim 12, lines 5 and 6, states that "pages" are "selectively displayed . . . by a . . . browser." That is, the "browser" determines which pages are displayed.

This makes sense at an SST or ATM: the "browser" determines which pages to display to the customer.

That "selective" display by the "browser" has not been shown

09/704,066  
Art Unit 2176  
9399.00

in Shima. In fact, the part of Shima relied on by the PTO appears to show the **opposite**.

The Final Action, page 4, second bullet-paragraph (beginning with "-A user interface . . .") asserts that a user interface having basic input and display capabilities shows the recitation in question. However, this assertion implies that the **user** selects which virtual control panel is displayed. Thus, the browser in Shima does not determine which virtual control panel is displayed.

This is consistent with the overall operation of Shima. The virtual control panel controls a specific device, and the user (not the browser) chooses which device to use, thereby choosing the virtual control panel to use.

#### **Claim 15**

The rejection of claim 15 asserts that the mere presence of one single supposed "passthrough object" in Shima shows claim 15. That is not so.

Claim 15 recites an examination step, to determine whether such an object already exists. Shima does not perform that step.

#### **Claim 17**

Claim 17 recites an "initiating page" and a "passthrough object." Those two **separate** entities have not been shown in Shima. As explained above, the PTO treats a **single** entity as both.

09/704,066  
Art Unit 2176  
9399.00

For example, the PTO treats the display in SKETCH 4, above, as both the "initiating page" and the "passthrough object."

#### Claim 18

Claim 18 recites a "register or established pages." That has not been shown in the Shima.

#### Claims 19 and 20

Dependent claim 19 states:

. . . wherein said passthrough includes a buffer for buffering events to be returned to pages that are no long established within the process.

That is, claim 19 states that, when some pages are no longer established, the buffer holds events for those pages.

Those pages, and the buffering, have not been identified in Shima.

This applies to claim 20.

Dependent claim 20 states: ". . . buffered events are returned to non-established pages after said pages have been re-established."

Establishment, and non-establishment, of pages has not been shown in Shima, nor the claimed return of events, after re-establishment.

09/704,066  
Art Unit 2176  
9399.00

GROUP 3: CLAIMS 21 - 27

Claim 21

The discussion of claim 1, above, applies to claim 21. This discussion is hereby incorporated by reference into this section.

Point 1

Claim 21 recites a computer-readable medium which contains instructions which perform specific functions.

That has not been shown in Shima. Shima's overall system does not correspond to such a "medium," even if his system does perform all the claimed functions, which is not so.

Point 2

Claim 21, second sub-paragraph, states that "pages" are "selectively displayed . . . by a . . . browser." That is, the "browser" determines which pages are displayed.

This makes sense at an SST or ATM: the "browser" determines which pages to display to the customer.

That "selective" display by the "browser" has not been shown in Shima. To repeat, no attempt has been made by the PTO to show the recitation in question.

09/704,066  
Art Unit 2176  
9399.00

Point 3

Claim 21 states that a "browser" runs on a computer. The "browser" creates the "pages," in which are "embedded" "control objects." The "control objects" initiate "calls" which reach "operational objects."

That is simply not found in Shima.

As explained above, Shima states that his computer can surf the Internet. (Column 18, lines 34 - 37.) However, that computer then becomes a **controlled device**. It replaces the VIDEO RECORDER in SKETCH 4, above.

That computer does not display the "control objects" discussed in the first paragraph of this Point 4. Nor do any such "control objects" initiate "calls" which reach "operational objects."

Point 4

In continuing Point 3, Appellant points out that the Final Action asserts that Shima's virtual control panels contain the "control objects." But the computer in Shima which surfs the Internet, and which the PTO thus asserts contains the claimed "browser," does not display any "control objects."

That computer does not control any devices in Shima's system.

**Claim 22**

Claim 22 states that computer "instructions" execute a

09/704,066  
Art Unit 2176  
9399.00

"process" which "creates" a "passthrough object."

The Final Action has not shown this. At best, the Final Action has asserted that Shima shows "pass-through functionality." (Final Action, page 8, second paragraph.)

Claim 22 does not recite that. "Pass-through functionality" does not correspond to the claimed creation of a "passthrough object."

#### Claim 24

Claim 24 recites an "initiating page" and a "passthrough object." Those two **separate** entities have not been shown in Shima. As explained above, the PTO treats a **single** entity as both.

For example, the PTO treats the display in SKETCH 4, above, as both the "initiating page" and the "passthrough object."

#### Claim 25

Claim 25 recites a "register or established pages." That has not been shown in the Shima.

#### Claims 26 and 27

Dependent claim 26 states:

. . . said passthrough object includes a buffer for buffering events to be returned to pages that are no long established within the process.

09/704,066  
Art Unit 2176  
9399.00

That is, claim 26 states that, when some pages are no longer established, the buffer holds events for those pages.

Those pages, and the buffering, have not been identified in Shima.

This applies to claim 27.

Dependent claim 27 states: ". . . buffered events are returned to non-established pages after said pages have been re-established."

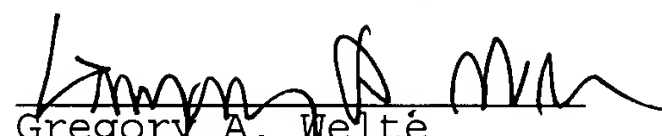
Establishment, and non-establishment, of pages has not been shown in Shima, nor the claimed return of events, after re-establishment.

09/704,066  
Art Unit 2176  
9399.00

# CONCLUSION

Appellant requests that the Board reverse all rejections, and pass all claims to issue.

Respectfully submitted,

  
Gregory A. Welte  
Reg. No. 30,434

NCR Corporation  
1700 South Patterson Blvd.  
WHQ - 3  
Dayton, OH 45479  
June 18, 2007  
(937) 445 - 4956

WELTE DIRECT: (765) 296 - 4699

ATTACHMENTS: -- CLAIMS APPENDIX  
-- STATEMENT THAT NO EVIDENCE APPENDIX IS PRESENT  
-- STATEMENT THAT NO RELATED PROCEEDINGS APPENDIX  
IS PRESENT

## 8. CLAIMS APPENDIX

1. Data processing apparatus having processing means, memory means and display means, wherein

said processing means performs a process in response to program instructions read from said memory means via dynamically linked operational objects called by control objects, such that events are returned back to a calling control object;

a plurality of pages are defined in a mark-up language that are selectively displayed and executed by a controlled browser;

said controlled browser is controlled by a controlling container object;

active control objects for calling said operational objects are contained within said container object;

a single pass-through object is created;

at least one of said pages includes a page embedded control object configured to call said passthrough object;

an initiating one of said page embedded objects calls said passthrough object and passes to said passthrough object output information detailing a desired call to a specified operational object;

said passthrough object interprets output information received from a page embedded object to generate a call to a contained object that in turn calls the desired operational object; and

said passthrough object receives event data from a called

09/704,066  
Art Unit 2176  
9399.00

operational object and returns input data to said initiating embedded object indicative of said returned event.

2. Apparatus according to claim 1, configured as a self service terminal.

3. Apparatus according to claim 2, wherein said self service terminal dispenses money and facilitates financial transactions.

4. Apparatus according to claim 1, wherein said mark-up language is hypertext mark-up language (HTML).

5. Apparatus according to claim 1, wherein said passthrough object is a separate process executed under the control of an operating system.

6. Apparatus according to claim 5, wherein any process configured to create a passthrough object will firstly ensure that such an object is not in existence and only create the object if it is not in existence thereby ensuring that only one passthrough object exists at any one time.

7. Apparatus according to claim 1, wherein said call to a contained object is made via a specific decoding object within said container object.

09/704,066  
Art Unit 2176  
9399.00

8. Apparatus according to claim 1, wherein return events are returned to an initiating page via said passthrough object.

9. Apparatus according to claim 8, wherein said passthrough object maintains a register of established pages.

10. Apparatus according to claim 9, wherein said passthrough object includes a buffer for buffering events to be returned to pages that are no long established within the process.

11. Apparatus according to claim 10, wherein buffered events are returned to non-established pages after said pages have been re-established.

12. A method of defining a process in a computer system via a plurality of pages defined in a mark-up language that are executable by a browser, in which

said process implements operations via dynamically linked operational objects called by control objects such that events are returned back to a calling control object;

a plurality of pages defined in a mark-up language are selectively displayed and executed by a controlled browser;

said controlled browser is controlled by a controlling container object;

active control objects for controlling said operational objects are contained within said container object;

09/704,066  
Art Unit 2176  
9399.00

a single passthrough object is created; and  
pages include a page embedded control object configured to call said passthrough object, said method comprising the steps of:

calling said passthrough object from an initiating embedded object and passing output information detailing a desired call to a specified operational object;

interpreting said output information at said passthrough object to effect a call to said operational object; and

returning input data to an initiating embedded object indicative of a returned event via said passthrough object.

13. A method according to claim 12, wherein said mark-up language is hypertext mark-up language (HTML).

14. A method according to claim 12, wherein said passthrough object is a separate process executed under the control of an operating system.

15. A method according to claim 14, wherein any process configured to create a passthrough object will firstly ensure that such an object is not in existence and only create the object if it is not in existence thereby ensuring that only one passthrough object exists at any one time.

16. A method according to claim 12, wherein said call to a

09/704,066  
Art Unit 2176  
9399.00

contained object is made via a specific decoding object within the container.

17. A method according to claim 12, wherein return events are returned to an initiating page via said passthrough object.

18. A method according to claim 17, wherein said passthrough object maintains a register of established pages.

19. A method according to claim 18, wherein said passthrough includes a buffer for buffering events to be returned to pages that are no longer established within the process.

20. A method according to claim 19, wherein buffered events are returned to non-established pages after said pages have been re-established.

21. A computer-readable medium having computer-readable instructions executable by a computer such that, when executing said instructions, said computer will perform the steps of:

establishing a library of dynamically linkable objects that may be called by control objects such that events are returned back to a calling control object;

establishing the availability of a plurality of pages defined in a mark-up language that may be selectively displayed and executed by a controlled browser, wherein said controlled browser

is controlled by a controlling container object;

containing active control objects for controlling operational objects within said container object; and

facilitating the establishment of a single passthrough object, wherein pages defined in said mark-up language include a page embedded control object configured to call said passthrough object such that, during a session with a user, said program computer will further perform the steps of:

calling said passthrough object from an initiating embedded object and passing output information detailing a desired call to a specified operational object;

interpreting said output information at said passthrough object to effect a call to said operational object; and

returning input information to an initiating embedded object indicative of a returned event via said passthrough object.

22. A computer-readable medium having computer-readable instructions according to claim 21, such that when executing said instructions, said computer will execute a process configured to create a passthrough object, wherein said process will firstly ensure that such an object is not in existence and only create the object if it is not in existence thereby ensuring that only one passthrough object exists at any one time.

23. A computer-readable medium having computer-readable instructions according to claim 21, such that when executing said

09/704,066  
Art Unit 2176  
9399.00

instructions, said computer will make a call to a contained object, wherein this call is made via a specific decoding object within the container.

24. A computer-readable medium having computer-readable instructions according to claim 21, such that when executing said instructions, a computer will return events to an initiating page via said passthrough object.

25. A computer-readable medium having computer-readable instructions according to claim 24, such that when executing said instructions, said computer will ensure that said passthrough object maintains a register of established pages.

26. A computer-readable medium having computer-readable instructions according to claim 25, such that when executing said instructions, a computer will ensure that said passthrough object includes a buffer for buffering events to be returned to pages that are no longer established within the process.

27. A computer-readable medium having computer-readable instructions according to claim 26, such that when executing said instructions, a computer will ensure that buffered events are returned to non-established pages after said pages have been re-established.

09/704,066  
Art Unit 2176  
9399.00

**9. EVIDENCE APPENDIX**

None.

09/704,066  
Art Unit 2176  
9399.00

**10. RELATED PROCEEDINGS APPENDIX**

None.